



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Self-organisation of Spatial Behaviour in a Kilobot Swarm

Citation for published version:

Imrie, C & Herrmann, JM 2017, Self-organisation of Spatial Behaviour in a Kilobot Swarm. in M Mangan, M Cutkosky, A Mura, PFMJ Verschure, T Prescott & N Lepora (eds), *Biomimetic and Biohybrid Systems: 6th International Conference, Living Machines 2017, Stanford, CA, USA, July 26--28, 2017, Proceedings*. Lecture Notes in Computer Science, vol. 10384, Springer International Publishing, Cham, pp. 551-561, The 6th International Conference on Biomimetic and Biohybrid Systems, Stanford, California, United States, 25/07/17. https://doi.org/10.1007/978-3-319-63537-8_49

Digital Object Identifier (DOI):

[10.1007/978-3-319-63537-8_49](https://doi.org/10.1007/978-3-319-63537-8_49)

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Peer reviewed version

Published In:

Biomimetic and Biohybrid Systems

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



Self-Organisation of Spatial Behaviour in a Kilobot Swarm

Calum Imrie and J. Michael Herrmann
Institute for Perception, Action and Behaviour
School of Informatics, University of Edinburgh
10 Crichton St, Edinburgh EH9 8AB, U.K.

July 28, 2017

Abstract

Applications of robotic swarms often face limitations in sensing and motor capabilities. We aim at providing evidence that the modest equipment of the individual robots can be compensated by the interaction within the swarm. If the robots, such as the well-known Kilobots, have no sense of place or directionality, their collective behaviour can still result in meaningful spatial organisation. We show that a variety of patterns can be formed based on a reaction-diffusion system and that these patterns can be used by the robots to solve spatial tasks. In this contribution, we present first results for applications of this approach based on ‘physically realistic’ Kilobot simulations.

1 Introduction

The design and control of robot swarms is often inspired by biological systems. However, simple, inexpensive robots may lack the sensory or motor capabilities of their intended biological counterparts such that following patterns that realise behavioural goals would not be achievable by individual robots. It is an interesting option to use the interaction among the robots as a source of information such that the swarm dynamics compensates the limitations of the individuals. In this way it may also be possible to reduce the effects of obstruction and interference amongst the robots. Related phenomena have been studied in social insects and even single living cells, where, as a paradigm, simple local rules lead to complex behavioural patterns in the swarm, which can enable decision making and improve efficiency [8].

Here we consider the Kilobot [9], a popular robotic swarm platform, which is a three-pronged robot with the two back prongs mechanically connected to vibrational motors. The Kilobot can sense ambient light by a single sensor, and it can send and receive messages by means of IR signals to other robots in the

immediate neighbourhood. Ref. [3] uses diffusive information to navigate Kilobots for random walks and shows that this approach allows control to improve the exploration behaviour by tuning the parameters to optimise either in an open or a closed environment.

We present here first results for a robot swarm controlled by a reaction-diffusion (RD) system which is studied for a collective of Kilobots in a few simple decision-making tasks. The Kilobot's lack of directionality suggests the use of this approach, because the RD equations (3) do not involve gradients with respect to spatial variables, and the diffusion terms do not require a directional comparison of potentials carried by neighbouring robots. The next section describes the pattern formation algorithm, which will be followed by the experimental setup, results and finally a discussion of applications and future work within the approach.

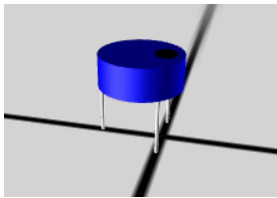


Figure 1: A Kilobot simulated in ARGoS.

2 Turing patterns

Reaction-diffusion (RD) systems produce stripe-like or honeycomb-like Turing patterns [10] if two substances are spreading with different diffusion constants such that the inhibitor is vanishing faster than the activator that in turn has caused its production. The RD equations

$$\begin{aligned}\frac{\partial u}{\partial t} &= D_u \Delta u + f(u, v) \\ \frac{\partial v}{\partial t} &= D_v \Delta v + g(u, v)\end{aligned}\tag{1}$$

describe the spatiotemporal dynamics of an activator u and an inhibitor v . The Laplace operator, $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$, represents the diffusion of u and v , typically with diffusion constants $D_u < D_v$. The dynamics can be realised e.g. by substances in solution, but a large variety of natural systems that follow this dynamics is known [1]. The functions $f(u, v)$ and $g(u, v)$ are the reaction models for the respective potentials and may vary for the different applications. Often, the activation function is non-linear and the inhibition is linear. We will use the FitzHugh-Nagumo model, i.e. choose the reaction terms, $f(u, v)$ and

$g(u, v)$, as

$$\begin{aligned} f(u, v) &= \lambda u - \alpha u^2 - (1 - \alpha)u^3 - \sigma v \\ g(u, v) &= u - v \end{aligned} \tag{2}$$

Whether the system produces spot-like or stripe-like patterns depends, resp., on the presence or absence of a quadratic term in the reaction models, which is governed in Eq. 2 by the parameter α . Moreover, dynamical patterns such as moving spirals are known to exist in RD systems.

The application of RD systems in robotics is just in its beginning. Morphogenetic robotics [6] uses Gene Regulatory Networks (GRN) in order to control multi-robot systems [5]. Also for the Kilobot problem attempts have been made to realise the potential of this approach. In principle, two options can be used: The robots can either realise the RD dynamics by estimating the robot concentration from the mutual distances and slow down, or speed up in order to change the pattern. We will follow here instead the simpler option to simulate the RD dynamics by exchanging messages with neighbouring robots to account for the diffusion in Eq. 1, while the behavioural consequences become apparent only after a pattern has formed. The clustering behaviour (Sect. 4.2, see also Refs. [2, 7] for earlier attempts) can be used as a precondition for the simulation, because it can gather all robots to enable a message exchange among all robots.

3 Experimental setup

The Kilobot swarm is simulated using ARGoS (see Fig. 1) using the Kilobot plug-in which includes their limited messaging capabilities. The Kilobot broadcast at each time step their u and v values and update their own values depending on the received messages. Due to limited messaging, each robot will receive a varying number of messages per time step. This is taken into account in the redefinition of the Laplace operator (1) for a discrete set of robots:

$$\Delta u = -ru_0 + \sum_{i=1}^r u_i \tag{3}$$

The sum runs over all neighbours of the robot and compares their potential with its own, u_0 . The operator (3) is stochastic, because the robot configuration that sends messages within a time step will typically deviate from a regular grid. It is possible to scale the diffusion constants in order to reduce the effect of the variable distances between neighbours, but we prefer Eq. 3 as it tends to produce a pattern as if the robots were in a regular formation, rather than relative to the embedding space. In addition to the messaging limitation, the messages themselves use limited information, namely 8 bit per time step, to implement the diffusive interaction. This can become problematic if small values of the potentials are to be broadcasted.

In addition, the RD dynamics needs to be tolerant to discretisation effects. In our ARGoS simulation, 225 Kilobots are run, usually in a 15×15 grid, with

an initial distance between the robots such that a communication with eight immediate neighbours is possible. A border is present so that when motion is introduced to the Kilobots they do not disperse completely, i.e. given enough time they will encounter another Kilobot. The initial values for the potentials were randomly chosen from $[0, 0.1]$. Uniform noise with a range $[-0.0005, 0.0005]$ was added to u to escape from chimeric states. Larger noise in combination with the discretisation noise may result in unstructured patterns, or transitions across multiple ground states which may impede subsequent decision making.

4 Results

4.1 Pattern formation in stationary swarm

Fig. 2 shows an example run of the set of simulations ran for a stationary swarm, and that a spotted pattern can emerge. Fig. 3 displays the location of the Kilobots as well as their activation values. The frequency of the spots in the pattern can be controlled through D_u , which is demonstrated in Fig 4. The spots will also remain stable over a long period of time, see Fig. 5.

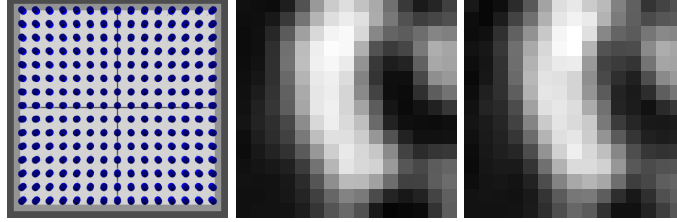


Figure 2: The simulation is of the stationary setup described in section 3. The first image is of the setup itself. The second and third images are the Kilobots' u and v values, resp., at the end of the simulation. These values were normalised to clearly see the differences between highly and weakly 'activated' Kilobots. In this case the largest values for u and v were, resp., 0.078 and 0.068. It can be seen quite clearly that there is a single spot, as to be expected since there is the inclusion of a quadratic term which will make the system favour spotted patterns. Parameters: $\alpha = 1$, $\sigma = 1$, $D_u = 0.5$, $D_v = 1.0$ and $\Delta t = 0.2$.

The type of pattern can also be modified, and it is possible to create stripe-like patterns within these simulations. Under the same setup as before but through modifying α and D_u the system can prefer striped patterns, which is shown in Fig 6. It does, however, take a significantly longer time for stable stripes to form in comparison to spot pattern formation. Once formed though, the pattern will remain stable, see Fig. 7.

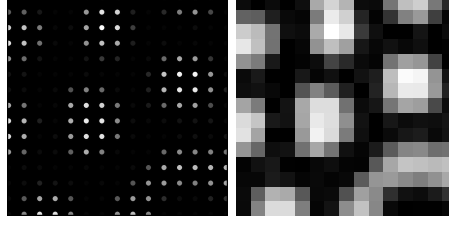


Figure 3: Example of a robot configuration in the physical world space. The images on the right are showing each Kilobot’s activation and inhibition, resp., and each circle represents the actual Kilobot’s place in the boxed environment, which is shown in Fig. 2.

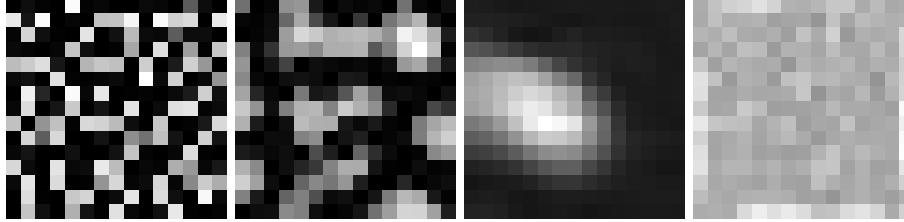


Figure 4: The simulations demonstrates the effect of increasing D_u . After the simulations the values were normalised, and the images left of the graph shows the Kilobots’ u values. As D_u increases, getting closer to the value of D_v , the spotted pattern becomes less spatially frequent, and also each spot becomes larger in size. Parameters: Similar to that in Fig. 2, except $D_u = 0.1, 0.3, 0.7$ and 0.9 (from left to right).

4.2 Clustering

A nearly regular configuration of the swarm can be achieved if the robots follow a form of preferential attachment where detachment is also allowed. The system is setup so that it favours spots, and after stable spots are formed a threshold value is used to determine the role of the Kilobot. The Kilobot’s with high activation will be the centre of these clusters, and the weakly activated will converge towards the highly activated. Let’s denote these roles as r_a and r_m , resp..

The adapt their speed by multiplication dependent upon the number of neighbours they hear from, and their respective roles. The power sent to the motors for r_a and r_m , denoted as p_a and p_m , are updated as follows

$$\begin{aligned} p_a &= p_a a_1^{n_a} a_2^{n_m}, \\ p_m &= p_m m_1^{n_a} m_2^{n_m}, \end{aligned} \quad (4)$$

where n_a and n_m are the number of messages received by a Kilobot with the role r_a and r_m respectively, and a_1, a_2, m_1 , and m_2 are constants. $a_2 > 1 > a_1 > 0$

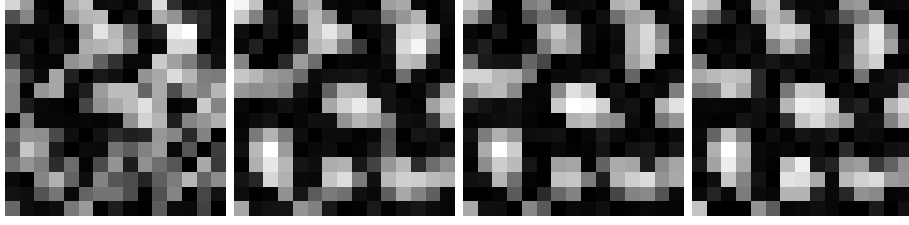


Figure 5: Stationary simulation where each image displays the normalised u values. The images are after 100, 400, 700 and 1000 time steps. The evolution of the spotted formations began as a larger spot which over time broke apart to form smaller stable spots. Parameters: Similar to that in Fig. 2 except $D_u = 0.3$.

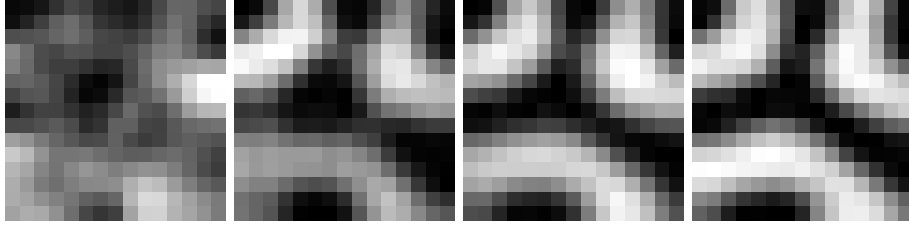


Figure 6: The results here show a simulation where $D_u = 0.55$, $D_v = 1.0$, $\Delta t = 0.02$, $\alpha = 0$ and $B = 1$. All other parameters are set as that in Fig. 2. The images displays the activation, u , of the simulated Kilobots. From left to right, images show time steps 500, 2000, 3000 and 4000. A pattern containing curved stripes is produced around time step 1000 and is then solidified by time step 4000.

and $m_2 > 1 > m_1 > 0$. Random motion is generated for both r_a and r_m . The power can never be below a minimum value, p_{min} . Kilobots will speed up if they do not hear from a r_a , and slow down if they do. If r_m is within a certain distance of a message sent from r_a , then they will change roles to r_a . This will further encourage stable clustering.

The constants in Eq. 4 will determine several aspects, such as compactness. The Kilobots have the capability to push a single Kilobot, and so if m_1 is set to be a gradual decline it will promote compactness. The Kilobots will converge on those that are highly active, as they will be surrounded by r_a s themselves, and thus will not increase their speed. When Kilobots of role r_m join the clusters and become r_a , then there is more potential for the Kilobots to receive messages from r_a , and remain as a cluster, see Fig. 8.

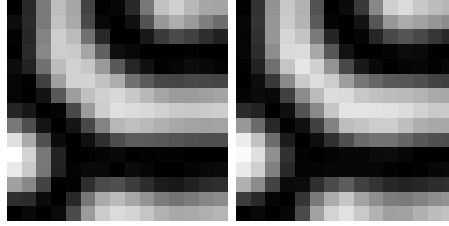


Figure 7: A simulation with the same setup as in Fig 6 showing the u values after 50000 and 600000 time steps. It can be seen that the pattern, once formed, remains fixed.

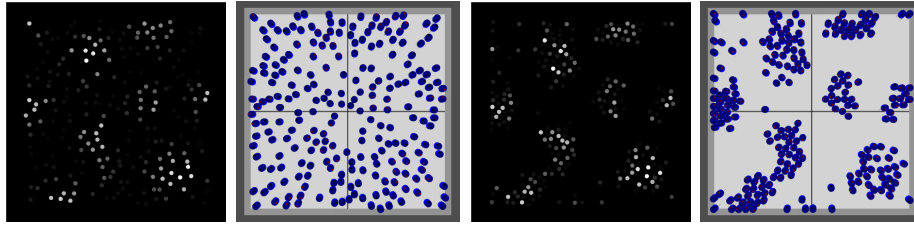


Figure 8: The Kilobots were randomly spatially distributed and only sent and received messages for the first 2500 time steps. If a Kilobot's u exceeds a threshold θ , then they are r_a and their LED turns red. Else they are r_m . Afterwards, the Kilobots will speed up if they do not receive messages from r_a , and slow down otherwise. The top two images displays the simulation after 2500 time steps, and the bottom two from time step 4000. Parameters: $D_u = 0.6$, $D_v = 1.0$, $\lambda = 0.97$, $\alpha = 1$, $\theta = 0.04$, $p_{min} = 0.01$, $a_1 = 0.1$, $a_2 = 1.1$, $m_1 = 0.9$, $m_2 = 1.1$, and $\Delta t = 0.2$.

4.3 Stripe segmentation

The Turing pattern constructed by the robots can be controlled to some extent. The wavelength of the striped pattern can be modified through the parameter D_u . This would lead to the swarm of having the ability to distinctly separate itself into subswarms. In this experiment, D_u was set to form at most two full parallel stripes through the swarm. Kilobots that have activations u below a certain threshold, would begin to randomly move until they arrive near other robots, then they slow down.

Once the stripes are formed the Kilobots will cluster into distinct groups. Fig. 9 displays an example simulation of this. In this particular case the a stripe was curving, meaning that the other stripe would barely form as it was parallel to this stripe. The initial layout of the Kilobots have them all facing forward. Though the stripes can form in different directions, which will result in different cluster formations, the stripes will always remain still, thus the other Kilobots will tend to stop around them as they will be consistently slowing down within the stripes presence.

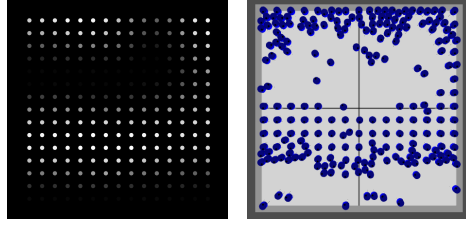


Figure 9: The stripes here were formed within 15000 time steps (left). After the formation of the pattern, the Kilobots with low u values randomly moved and would slow down when in contact with other Kilobots, which can be seen after 370 time steps of stripes formation (right). Parameters: $D_u = 0.63$, $\alpha = 0$ and $\Delta t = 0.5$.

4.4 Ring formation

It is also possible for the Kilobots, to some success, to form rings based on the information provided by the RD system. Similar to that in the stripe segmentation procedure, the Kilobots initially act solely as a platform for the RD system. The system is setup so that it favours spots, and after stable spots are formed a threshold value is used to determine the role of the Kilobot. These roles will be denoted, as that in section 4.3, r_a and r_m . Kilobots with the role of r_a will form the shape of the ring, and r_m will fill in the gaps.

Random motion is generated for r_m while for r_a there is a probability of 0.2 to perform a random movement, while otherwise the movement depends on the messages received. If r_a receives a message from another active robot r'_a then it shall turn left, otherwise it will move forward. Due to the lossy communication, it is unlikely that both of the two active Kilobots will receive a message from each other on the same time step, and thus will result in a repulsion force. They follow the same power updates as that in Eq. 4, but here $a_1 > 1 > a_2 > 0$.

The setup described above takes advantage of the spotted pattern created since the spots, i.e. r_a , will immediately repel each other to create a loose ring. The other Kilobots will then, through random movement, converge onto these loose structures. The size of the ring is controlled through a_1 and a_2 , as this determines the initial repelling force within each of the spots produced.

The majority of spots will lead to ring formations, though difficulty arises with spots that are present at the borders. These will mostly transform into semi-rings and more cluster like groupings. Fig. 10 displays an example simulation on a randomly spatially distributed swarm.

5 Discussion

This work has demonstrated not only are Turing Patterns possible in a Kilobot swarm, even with their communication difficulties, pattern formation can be decided upon (spots or stripes) through parameter choice, and how these

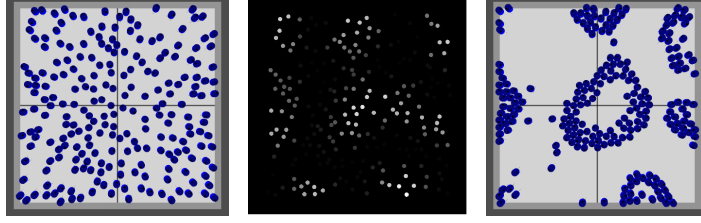


Figure 10: The Kilobots were initially randomly spatially distributed and were stationary for the first 10^4 time steps, which is seen in the left image. The spots formed after this period can be seen in the middle. The Kilobots were then given a role dependent upon u . If $u > \theta$ then they are r_a , else r_m . r_m would move randomly and will slow down if they hear from a r_a , and speed up otherwise. r_a will speed up and turn left if they hear from a r_a , and slow down and move forward otherwise. r_a also has a small chance of moving randomly. Parameters: $D_u = 1.1$, $D_v = 2.0$, $\alpha = 0$, $\lambda = 0.97$, $\theta = 0.135$, $a_1 = 1.8$, $a_2 = 0.8$, $m_1 = 0.2$, $m_2 = 1.05$, and $\Delta t = 0.05$.

patterns form, such as the number of spots. There had been work conducted into the sensitivity of the parameters, in particular how the parameters affect the pattern. One of the main parameters is the diffusion ratio between D_u and D_v , and this can be used as a control parameter for the pattern (spots, stripes; a third type, inverted spots, was not considered here).

This ability to produce stable patterns can then lead to various applications. The spotted pattern, see Fig. 5, can be controlled with respect to the number of spots, and this produces distinct teams within the swarm. This would be ideal for team-based tasks, including swarm exploration as this would allow groups of robots to detach from the swarm and conduct their exploration within their own group.

The stripe segmentation used a very basic rule for velocity. It was to show how the stripes can separate a swarm and that the wavelength can be controlled to determine the number of splits. Once the stripes have been formed then it is relatively simple to have control of the swarm by allocating the movement to those below/above a threshold that can be found analytically. The Kilobots can also communicate the maximum value of the potential during formation of a pattern, and use this information to determine the threshold (e.g. as half the maximum). This can also split them into types which again can be communicated to each other, and thus have more specific segmentation behaviours. For example, all non active Kilobots should avoid active Kilobots and form their own cluster.

The other formation demonstrated was the ring formation. The wall borders hinder ring formation, but as stated in section 3, the borders contain the Kilobots' Brownian motion and eliminate the chance of swarm dispersement. One solution to this would be the use of lighting, which the Kilobots could detect and move away from. This would provide a non-obstacle repellent that would

keep the Kilobots contained and potentially increase the success of ring formation. Another approach would be to change the Kilobots' motion so that it can home in early on either Kilobots that have the role r_a or have seen r_a . While this is difficult given the lack of directional information, it is possible to gain this information using past data. Interestingly this is where the RD system can provide additional information such as diffusivity, which will allow the Kilobots to move with the diffusion, thus a smoother spatial transition.

In order to realise the experiments with real Kilobots, finite battery life needs may be problematic. Obtaining straight stripes (Fig. 9), would be around two hours real time, which considerably exceeds the typical battery life of 20 minutes, although in most of our experiments the robots move only for part of the time. Therefore other robotic platforms may be preferable. A more capable robotic swarm would also have improved communication abilities, and thus reduce the time of pattern formation due to lossy communication.

Critical behaviour within a swarm can lead to improved results within a swarm, and work has been conducted into parameter selection specifically for particle swarm optimisation [4]. Turing patterns themselves require the correct parameters to be chosen to be able to form, thus they too will have a critical point w.r.t. the particular pattern formed. The critical behaviour in this swarm would allow the swarm to easily transition between one pattern to the other, thus increasing the range of information and behavioural capabilities of the entire swarm.

6 Conclusion

This investigation has demonstrated through preliminary analysis that the Kilobots are capable of constructing Turing patterns via a RD system through message passing. Furthermore, they have the capability of creating both spotted and striped patterns with control of how the patterns will be formed. This could lead to applications such as swarm separation and team formation. Through the information from the final pattern that was formed, the Kilobots have the capability to cluster and to form teams and boundaries. Although the Turing patterns offer only a few formation types, further work will show that a larger manifold of behaviours will be achievable by scheduling or adapting the parameter values or in combination with other techniques.

References

- [1] J. A. Adam. *Mathematics in nature: Modeling patterns in the natural world*. Princeton University Press, 2006.
- [2] N. Correll and A. Martinoli. Modeling and designing self-organized aggregation in a swarm of miniature robots. *Int. J. Robotics Research*, 30(5):615–626, 2011.

- [3] C. Dimidov, G. Oriolo, and V. Trianni. Random walks in swarm robotics: An experiment with Kilobots. In *Int. Conf. Swarm Intelligence*, pages 185–196, 2016.
- [4] A. Erskine, T. Joyce, and J. M. Herrmann. Parameter selection in particle swarm optimisation from stochastic stability analysis. In *Int. Conf. Swarm Intelligence*, pages 161–172. Springer, 2016.
- [5] H. Guo, Y. Meng, and Y. Jin. A cellular mechanism for multi-robot construction via evolutionary multi-objective optimization of a gene regulatory network. *BioSystems*, 98(3):193–203, 2009.
- [6] Y. Jin and Y. Meng. Morphogenetic robotics: An emerging new field in developmental robotics. *IEEE Transact. Systems, Man, Cybern. C*, 41(2):145–160, 2011.
- [7] W. Li, M. Gauci, and R. Groß. Turing learning: A metric-free approach to inferring behavior and its application to swarms. *Swarm Intelligence*, 10(3):211–243, 2016.
- [8] H. Oh, A. R. Shirazi, C. Sun, and Y. Jin. Bio-inspired self-organising multi-robot pattern formation: A review. *Robotics and Autonomous Systems*, 2017.
- [9] M. Rubenstein, C. Ahler, and R. Nagpal. Kilobot: A low cost scalable robot system for collective behaviors. In *IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 3293–3298, 2012.
- [10] A. M. Turing. The chemical basis of morphogenesis. *Phil. Transact. Royal Society London B*, 237(641):37–72, 1952.